



A quoi tu ressembles sans ton masque ?

Séance-Projet 3 : Reconnaissance et Reconstruction de Visages

N'oubliez jamais que vous avez à votre disposition 32 personnes et 6 postures

1 Rappel du contexte

Avec la pandémie de la Covid-19, nous avons appris à porter des masques quotidiennement. Désormais la moitié de notre visage est cachée nous laissant le regard pour seul moyen d'expression visible. Cette opération de cacher le visage s'apparente à un domaine en traitement d'images et de vidéos appelé "inpainting". En effet, les pixels correspondant à la zone du masque (modélisé par un rectangle) sont mis à 0. Et les *eigenfaces* permettent, entre autres, de restaurer la zone dégradée.

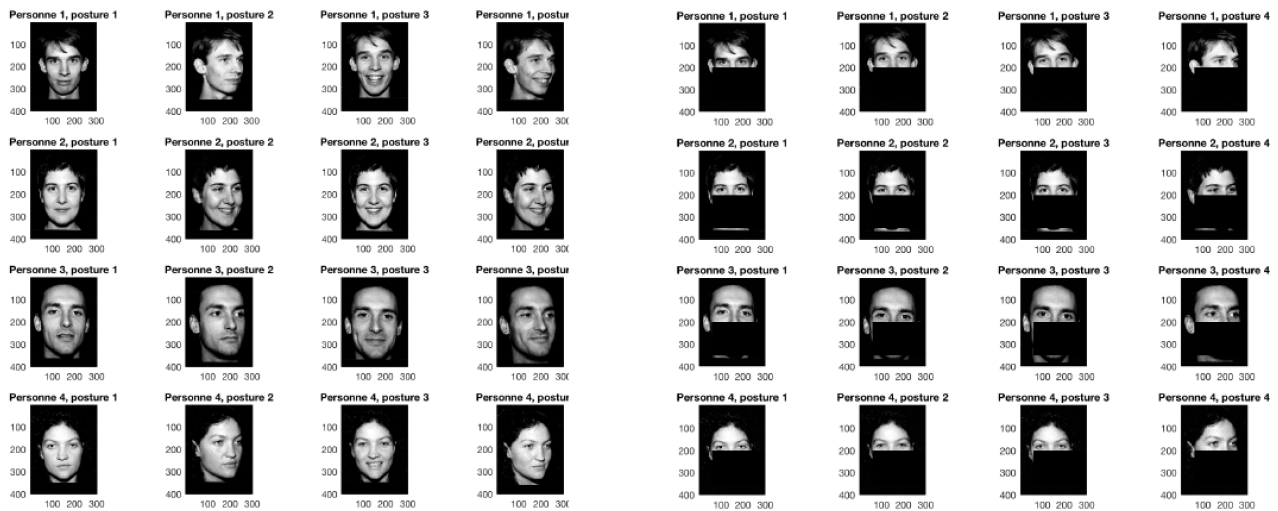


FIGURE 1 – Une base de visages et de visages masqués

Données manipulées

La base de données sur laquelle ce projet est appliqué est une collection d'images utiles pour mener des expériences en psychologie (*Psychological Image Collection at Stirling (PICS)*¹). Elle comporte 32 personnes (16 hommes et 16 femmes) avec 6 postures/expressions faciales (face, trois quart face et trois émotions différentes par posture). La FIGURE 1 montre une base de $n = 16$ visages présentant 4 personnes masquées et non masquées dans 4 positions/émotions différentes relativement à la caméra.

2 Retour sur les eigenfaces

Pour pouvoir remplir la zone du masque, nous nous appuyons sur le travail de M. Turk et A. Pentland intitulé «Eigenfaces for Recognition»² [1].

Il constitue une des applications les plus intéressantes et populaires de l'ACP au domaine de la reconnaissance de forme. Il s'agit tout simplement d'appliquer l'ACP à partir de données de très grandes tailles : des images de visages. Ici, l'ACP va nous permettre de réduire la dimension des images et de créer un système de reconnaissance grâce aux composantes principales pour retrouver le visage entier dans la base d'apprentissage le plus similaire au visage masqué et de permettre une reconstruction de la zone du masque (en faisant le lien avec le visage sans masque correspondant).

Pour réaliser cette reconnaissance, il est inutile et inopportun d'utiliser les $p = 120000$ niveaux de gris pour comparer l'image masquée avec chacune des images de la base d'apprentissage. L'ACP est donc un prétraitement qui consiste à réduire la dimension des données selon le schéma suivant où le prédicteur h prend en entrée la sortie de l'ACP :

$$\mathbf{x} \in \mathbb{R}^{120000} \rightarrow \boxed{\text{ACP}} \rightarrow \mathbf{c}(\mathbf{x}) \in \mathbb{R}^{q \ll 120000} \rightarrow \boxed{h} \rightarrow h(\mathbf{x})$$

Le vecteur caractéristique $\mathbf{c}(\mathbf{x})$ est tout simplement formé des q premières composantes principales résultant de la projection de l'image requête \mathbf{x} sur les q premiers vecteurs propres unitaires de la matrice Σ correspondant aux images masquées

Question 1 :

Le fichier `eigenfaces_part3.m` vous est fourni et réalise une partie du travail de la séance 1 (création de la base d'apprentissage, calcul des «eigenfaces» pour les visages sans masque, ...).

1. lisez bien le code pour comprendre le rôle des variables définies,
2. complétez le code pour calculer les «eigenfaces» des visages avec masque,
3. modifiez le code pour utiliser les fonctions de calcul de couples propres utilisant la méthode «Subspace Iteration» que vous avez développées à la partie 2 du Projet.

1. <http://pics.stir.ac.uk>

2. <https://www.mitpressjournals.org/doi/pdfplus/10.1162/jocn.1991.3.1.71>

3 Reconnaissance et reconstruction

3.1 Reconnaissance

A partir des composantes principales de la base des visages avec masque, on a défini ainsi $c(\mathbf{x})$, une représentation de \mathbf{x} , image requête (ou de test), dans cette base :

$$\mathbf{c}(\mathbf{x}) \stackrel{\text{déf}}{=} \begin{bmatrix} \mathbf{w}_1^\top \mathbf{x} \\ \mathbf{w}_2^\top \mathbf{x} \\ \vdots \\ \mathbf{w}_q^\top \mathbf{x} \end{bmatrix}$$

On dit que $\mathbf{c}(\mathbf{x})$ est une représentation compacte préservant au mieux l'information contenue dans \mathbf{x} avec seulement q composantes principales. La prédiction h peut alors être définie de diverses manières.

Parmi les possibilités les plus simples, on peut chercher les images \mathbf{x}_i de la base dont les représentations compactes sont les plus proches de celle associée à l'image requête \mathbf{x} , ce qui revient à minimiser la fonction $h(\mathbf{x})$ définie par :

$$h(\mathbf{x}) = \arg \min_{i \in 1 \dots n} d(\mathbf{c}(\mathbf{x}_i), \mathbf{c}(\mathbf{x}))$$

où $d(\mathbf{c}(\mathbf{x}_i), \mathbf{c}(\mathbf{x})) = \|\mathbf{c}(\mathbf{x}_i) - \mathbf{c}(\mathbf{x})\|_2$.

La fonction $h(\mathbf{x})$ est aussi appelée **classifieur**. Elle représente ici la méthode des 1-plus proche voisin vue lors du TP4 d'Analyse de Données. Ce classifieur peut être étendu à d'autres méthodes comme la version générale des k -plus proches voisins que vous pouvez réutiliser pour cette partie du projet.

Question 2 :

Le script `clusters.m` affiche sous la forme d'un nuage de n points de \mathbb{R}^2 les deux premières composantes principales des visages sans masques de la base d'apprentissage.

Chaque couleur correspond à une même personne de la *base d'apprentissage*. Ce nuage fait apparaître des groupes de points (ou *clusters*) de couleur uniforme, groupes plus ou moins séparés suivant les personnes que l'on sélectionne.

Avec les deux premières composantes, il semble cependant difficile de bien séparer les personnes. Il faudra sûrement rajouter d'autres composantes pour réussir cette séparation, nombre lié aux personnes, à leur nombre; ce nombre de composantes nécessaires est lié au pourcentage d'information qu'elles contiennent (contraste).

1. rajoutez dans le script `eigenfaces_part3.m`, le calcul des contrastes si on considère 1, 2, 3, ... «eigenfaces» (sans ou avec masque),
2. dans le script `clusters.m`, rajoutez l'affichage avec les 3 premières composantes (affichage dans \mathbb{R}^3).

Question 3 :

Le script `reconnaissance_sans_masque.m` tire aléatoirement une image de test, parmi les 32 personnes et les 6 postures faciales disponibles dans la base de données complète. Il travaille sur les visages sans masque et donc avec les «eigenfaces» correspondantes³

3. vous permettant ainsi de commencer à travailler sur cette question sans avoir fini de compléter le script `eigenfaces_part3.m`

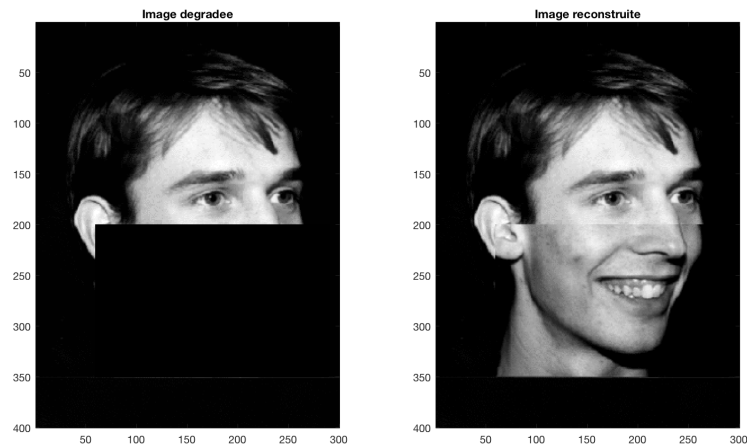
1. complétez le script pour trouver l'individu (personne+posture) dans la base d'apprentissage qui est le plus proche de l'image de test (vous pouvez utiliser et adapter la fonction `kppv` que vous avez écrite lors du TP4),
2. en recopiant le script dans un fichier `reconnaissance_avec_masque.m`, faites la reconnaissance avec masque,
3. identifiez les paramètres de votre classifieur et essayez de trouver des valeurs optimales à ces paramètres (sans ou avec masque).

3.2 Reconstruction

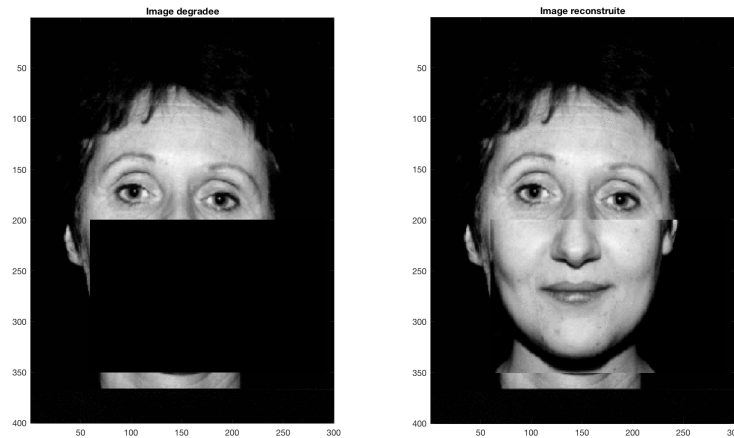
La reconstruction peut être réalisée en remplaçant la zone correspondant au masque par la zone de l'image de la base d'apprentissage de visages entiers la plus proche dans l'espace défini par les eigenfaces masqués.

Voici deux exemples de reconstruction d'images après reconnaissance :

- personne présente dans la base d'apprentissage mais avec une posture/émotion différente :



— personne ne se trouvant pas dans la base d'apprentissage :



Question 4 : reconstruction

En partant du script `reconnaissance_avec_masque.m` (que vous recopierez dans un fichier `reconnaissance_reconstruction.m`), développez le code permettant de reconstruire le visage de test.

Question 5 : évaluation

Pour évaluer la reconnaissance du classifieur, vous utiliserez les images reconstruites correspondant à plusieurs requêtes.

En réutilisant les outils qui vous ont été fournis ou que vous avez développés pour la partie 1 et cette partie 3, imaginez une évaluation de la reconstruction.

4 Documents à rendre

Voir sous Moodle

5 Bibliography

- [1] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *J. Cognitive Neuroscience*, 3(1) :71–86, 1991.

**N'oubliez jamais que vous avez à votre disposition
32 personnes et 6 postures**